

Remarks

Reconsideration of this application as amended is respectfully requested.

Claims 1-19 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,134,583 of Herriot ("Herriot") and U.S. Patent No. 5,636,355 of Ramakrishnan et al. ("Ramakrishnan").

Applicants respectfully submit that amended claim 1 is not obvious in view of Herriot and Ramakrishnan. Amended claim 1 is a virtual machine that includes the limitations

class structure for holding one or more of a set of predefined classes for use by an application program that executes under the virtual machine;

class loader that obtains one or more of the predefined classes via a network and that stores the predefined classes into a memory for holding the class structure;

memory manager that purges selected ones of the predefined classes from the class structure so as to minimize an amount of the memory consumed by the predefined classes in the class structure and to minimize class loading activities on the network.
(Amended claim 1) (Emphasis added).

Neither Herriot or Ramakrishnan disclose or suggest a virtual machine with a memory manager that minimizes an amount of the memory consumed by predefined classes as claimed in amended claim 1. Moreover, neither Herriot or Ramakrishnan disclose or suggest a virtual machine with a memory manager that minimizes class loading activities on a network as claimed in amended claim 1.

(1)
new
matter
minimize

Herriot discloses a system for obtaining classes from a server and storing the obtained classes in a cache file (Herriot, col. 11, lines 20-23) but does not teach or suggest a memory manager that minimizes an amount of memory consumed by classes and that minimizes class loading activities on a network as claimed in amended claim 1.

Ramakrishnan discloses a method for managing a disk

cache memory (*Ramakrishnan*, col. 2, lines 54-55) rather than a virtual machine with a memory manager as claimed in amended claim 1. Rather than minimizing an amount of memory consumed by classes and minimizing class loading activities on a network as claimed in amended claim 1, *Ramakrishnan* teaches

The present invention resides in a method, and related apparatus, for managing a disk cache memory that can be used to advantage in reducing the number of disk write accesses needed to satisfy disk write requests.

(*Ramakrishnan*, col. 2, lines 54-57) (Emphasis added). In further contrast to purging selected classes from a class structure so as to minimize an amount of the memory consumed by classes in the class structure and to minimize class loading activities on a network as claimed in amended claim 1, *Ramakrishnan* teaches that

The selection of blocks to purge to the disk may be based on a combination of factors, such as the number of dirty blocks in a cylinder or platter, the distance of the dirty blocks from the current head position, or the ages of the blocks since the last access.

(*Ramakrishnan*, col. 5 lines 50-54).

It is also submitted that *Herriot* does not teach or suggest a combination with *Ramakrishnan* and that *Ramakrishnan* does not teach or suggest a combination with *Herriot*. Instead, the teachings of *Herriot* are directed to a client computer which obtains files from a server computer (*Herriot*, col. 2, lines 17-28) and the teachings of *Ramakrishnan* are directed to managing a disk cache memory (*Ramakrishnan*, col. 2, lines 54-57).

It would be impermissible hindsight based on an applicant's own disclosure to incorporate the disk cache memory management teachings of *Ramakrishnan* into the client-server file loading teachings of *Herriot*. Moreover, any such combination would still lack a memory manager that minimizes an amount of the memory consumed

by classes and that minimizes class loading activities on a network as claimed in amended claim 1.

It is therefore respectfully submitted that the virtual machine of amended claim 1 which includes a memory manager that minimizes an amount of the memory consumed by classes and that minimizes class loading activities on a network is not obvious in view of *Herriot* which teaches caching of files obtained from a server and *Ramakrishnan* which teaches reducing the number of disk write accesses for a disk cache memory.

Given that claims 2-11 depended from amended claim 1, it is also submitted that claims 2-11 are not obvious in view of the referenced cited by the Examiner.

(2) It is also submitted that amended claim 12 is not obvious in view of *Herriot* and *Ramakrishnan*. Amended claim 12 is a method for class loading in a virtual machine that includes limitations similar to the limitations of amended claim 1 including purging classes from a class structure so as to minimize an amount of memory space consumed by the classes in the class structure and to minimize class loading activities on a network. Therefore the remarks stated above with respect to amended claim 1 also apply to amended claim 12.

It is therefore respectfully submitted that the method for class loading of amended claim 12 which minimizes an amount of the memory consumed by classes and minimizes class loading activities on a network is not obvious in view of the teaching of *Herriot* which are directed to caching files obtained from a server and the teachings of *Ramakrishnan* which are directed to reducing the number of disk write accesses for a disk cache memory.

Given that claims 13-15 depended from amended claim 12, it is also submitted that claims 13-15 are not obvious in view of *Herriot* and *Ramakrishnan*.

(3) It is further submitted that amended claim 16 is not obvious in view of *Herriot* and *Ramakrishnan*. Amended claim 16 is a device that includes limitations similar to the limitations of amended claim 1 including a memory manager that minimizes an amount of memory consumed by classes and minimizes class loading activities on a network. Therefore the remarks stated above with respect to amended claim 1 also apply to amended claim 16.

Given that claims 17-19 depended from amended claim 16, it is submitted that claims 17-19 are not obvious in view of *Herriot* and *Ramakrishnan*.

It is respectfully submitted that in view of the amendments and arguments set forth above, the applicable objections and rejections have been overcome.

The Commissioner is authorized to charge any underpayment or credit any overpayment to Deposit Account No. 08-2025 for any matter in connection with this response, including any fee for extension of time, which may be required.

Respectfully submitted,

Date: 8-28-02

By: _____

Paul H. Horstmann
Reg. No.: 36,167



COPY OF PAPERS
ORIGINALLY FILED

Version with Markings to Show Changes Made

1. A virtual machine, comprising:
 - class structure for holding one or more of a set of predefined classes for use by an application program that executes under the virtual machine;
 - class loader that obtains one or more of the predefined classes [from a network server] via a network and that stores the predefined classes into a memory for holding the class structure;
 - memory manager that purges selected ones of the predefined classes from the class structure so as to [optimize the use of] minimize an amount of the memory [resources] consumed by the predefined classes in the class structure and to minimize class loading activities on the network.
2. The virtual machine of claim 1, wherein the class loader obtains the predefined classes from [the network server is] an HTTP server that exports a set of class files containing one or more of the predefined classes.
10. The virtual machine of claim 7, wherein the memory manager purges the least recently used or the next recently used one of the predefined classes if an amount of available memory [resources] falls below a predetermined threshold level.
12. A method for class loading in a virtual machine, comprising the steps of:
 - obtaining one or more of a set of predefined classes [from] via a network [server];
 - storing the predefined classes into a class structure for use by an application program that executes

under the virtual machine;

purging selected ones of the predefined classes from the class structure so as to [optimize the use of] minimize an amount of memory space [resources] consumed by the predefined classes in the class structure and to minimize class loading activities on the network.

16. A device, comprising:

memory that holds a class structure for storing one or more of a set of predefined classes for use by an application program;

processor that executes the application program and a class loader that when executed obtains one or more of the predefined classes [from] via a network [server] and stores the predefined classes into the class structure for use when executing the application program, the processor further executing a memory manager that when executed purges selected ones of the predefined classes from the class structure so as to [optimize use of the memory] minimize an amount of the memory consumed by the predefined classes in the class structure and to minimize class loading activities on the network.